SOFTWARE ENGINEERING (CO-301)

MEDICANT

Disease Diagnosis and preventive care Application

1. Introduction

'MEDICANT' is an android based smart application that focuses on the preventive care and wellbeing of the users, endowed with a flexible interface that aims at facilitating the work of Doctors, the cornerstones of society. The system provides a platform where patients can book their appointments at their own convenience while adhering to the doctor's schedule. The system essentially emphasizes the importance of extensive communication between patients and doctors with the help of the in-app chatbox.

2. Rationale of the Study

Many deadly diseases are preventable and can be cured if treated at an early stage. However, people tend to ignore the early symptoms of a disease as they are mild. And till the time they become aware of the disease, the infection has already spread into their body. Hence, it is always advisable to have a regular body check-up to avoid health issues later. But due to the day-to-day busy schedule many times it is not possible for the patients to go and get a medical appointment booked in person. Which leads to a delay in the treatment and may cause a major health problem later.

The Application we are developing is an android based smart application that will help the users to identify a disease based on the symptoms allowing the users to seek out preventive care before it's too late. Then based on the disease identified, the user can book appointments with doctors in their vicinity specializing in the treatment of that particular disease. Users are able to view their profile which may further help them in reaching a decision.

The application is furnished with a well designed user-friendly interface that facilitates users to save time and increase productivity. Thereby, allowing the doctors to rationally allocate time and impart medical aid to more patients.

The application is further augmented with the help of a chatbox that enables smooth communication between doctors and patients which may prove to be quite useful in various circumstances.

3. Literature Review

The widespread adoption and use of software technologies is opening new and innovative ways to improve health and health care delivery.

E-health is the use of digital information and communication technologies, such as computers and mobile devices, to access health care services remotely and manage your health care.

Using technology to deliver health care has several advantages, including cost savings, convenience, and the ability to provide care to people with mobility limitations, or those in rural areas who don't have access to a local doctor or clinic. A study by Stephanie Watson,Executive Editor, Harvard Women's Health Watch (2020) illustrated the potential of e-health and how the use of it has grown significantly over the last decade.

E-health has become even more essential during the coronavirus (COVID-19) pandemic. Fears of spreading and catching the virus during in-person medical visits have led to a greater interest in, and use of, technology to provide and receive health care.

E-health provides a plethora of services and facilities ranging from recording your blood sugar levels, calorie intake etc. and being notified when you are due for an examination, vaccination or other preventive to virtual appointments with a specialist.

The applications of E-health can be further augmented to a massive scale by employing the labor of machine learning.

While it is needless to say, Machine learning has established its role in the healthcare industry with several notable contributions from robotic surgery to cancer detection and prediction, Considering the gargantuan amount of patient data that needs to be processed from different sources such as medical records, medical images, genomic data, mobile phone data, wearable devices (smartwatches and sensors), and more. It becomes imperative to essentially build a system that makes collecting, processing and extracting insights from this data less cumbersome with the help of machine learning. This ultimately made it possible to create smart electronic healthcare records which is yet another vital application of E-health that involves maintaining a centralized electronic health record system containing information ranging from the user's personal details and contact information to medical history, prior diagnoses, medications and drug allergies that assists in providing the user a personalized treatment plan.

Apart from this, Machine Learning has numerous techniques such as NLP, Convolutional Neural Networks(Deep Learning) that can ameliorate and upgrade ehealth.

NLP is used primarily to identify key elements (entities, intent, relationships) in free-form text and make that data usable by computer systems in various ways. Digital personal assistants like Siri, Cortana, and Alexa make heavy use of concepts in NLP to do what they do.

Similar to the recent explosion of personal assistant AI in the consumer market, the healthcare industry is seeing a particularly remarkable surge in apps that use NLP to give our devices and physicians a better understanding of our health. Healthcare chatbots are a prime example of this.

Chatbots are software developed with machine learning algorithms, including natural language processing (NLP), to stimulate and engage in a conversation with a user to provide real-time assistance to patients. Patients love speaking to real-life doctors, and artificial intelligence is what makes chatbots sound more human. In fact, some chatbots with complex self-learning algorithms can successfully maintain in-depth, nearly human-like conversations.

Machine learning applications are beginning to transform patient care as we know it. Although still in its early stages, chatbots will not only improve care delivery, but they will also lead to significant healthcare cost savings and improved patient care outcomes in the near future.

The advantages of using hybrid chatbots in healthcare are enormous – and all stakeholders share the benefits.

The article published by Kalinin(2020) substantiates the extent to which chatbots are economizing the healthcare industry. Chatbots are gradually reducing hospital wait times, consultation times, unnecessary treatments, and hospital readmissions by connecting patients with the right healthcare providers and helping patients understand their conditions and treatments even without visiting a doctor.

Furthermore, hospitals and private clinics use medical chatbots to triage and clerk patients even before they come into the consulting room. These bots ask relevant questions about the patients' symptoms, with automated responses that aim to produce a sufficient history for the doctor. Subsequently, these patient histories are sent via a messaging interface to the doctor, who evaluates the results and decides which patients need to be seen first and which patients require a brief consultation.(Kalinin, 2020)

As versatile as NLP is, it comes to standstill when confronted with data that is not text based such as MRI, X-RAY or other diagnostic images to process and draw insights from.The competence of deep convolutional neural networks helps us to tackle this situation.
CNNs were originally researched by D.H. Hubel and T.N. Wiesel who studied the way that mammals visually perceive the world using layered neuron architectures. The mammalian visual cortex visualises objects through a hierarchical structure of features that are formulated based on the input stimuli (such as oriented edges and spatial invariance). This inspired the construction of a similar system for computer vision. Part of the input image

that arrives through a receptive field and its data is iteratively passed through the layers. [1]

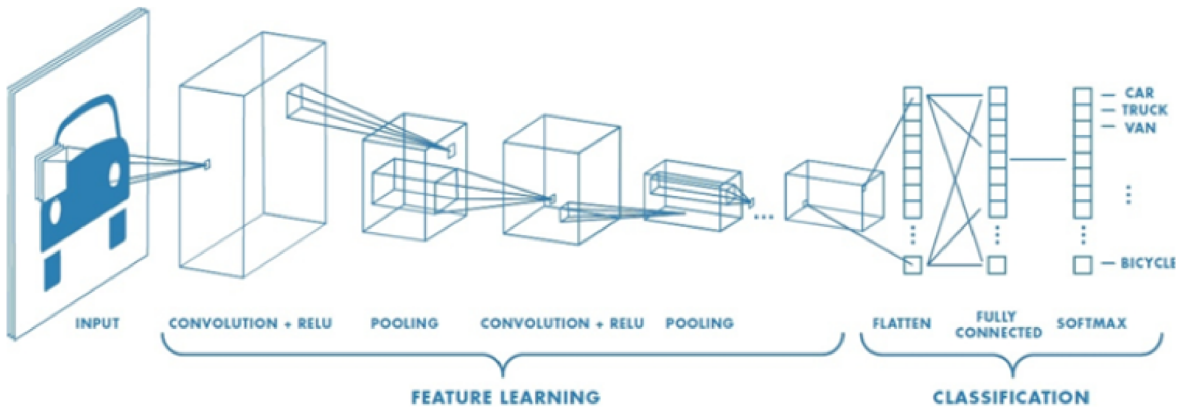


Fig. 1: Layers of CNN

With the help of CNNs, we are able to have a better understanding of medical images and help us in analyzing and classifying medical images while detecting anomalies outside of the network's training.

High resolution computed tomography (HRCT), Reverse Transcription Polymerase Chain Reaction (RT-PCR), Electrocardiogram (ECG), Mammography are a few examples of CNN's success and impact.

But it's not all x-rays and mammograms in the medical deep-CNN space. Companies like 'Face2Gene' are using facial recognition and images taken by smartphones to classify phenotypes and accurately diagnose developmental syndromes.

However, achieving such a subliminal level of accuracy in medical imaging is not simple. The article named 'Convolutional neural networks in medical image understanding: a survey' by Dr. R. Sarvamangala co-authored by Raghavendra V. Kulkarni provides an insight on the challenges faced while designing CNN's.

With the advent of new technologies in the field of medicine, large amounts of patient data have been collected and are available to the medical research community. However, the accurate prediction of a disease outcome is one of the most interesting and challenging tasks for physicians. As a result, ML methods have become a popular tool for medical researchers. These techniques can discover and identify patterns and relationships between them, from complex datasets, while they are able to effectively predict the disease.

A variety of these techniques, including Artificial Neural Networks (ANNs), Bayesian Networks (BNs), Support Vector Machines (SVMs) and Decision Trees (DTs) have been widely applied in the research and development of predictive models, resulting in effective and accurate decision making.

The research paper **'GDPS - General Disease Prediction System'** substantiates how an efficient model can be built based on the ID3 Decision Tree Algorithm by classifying the symptoms based on their information gain prevalent in the dataset.

And the article **'Disease Prediction System Using Fuzzy C-Means Algorithm'** authored by T. Bala Ramya illustrates how the Fuzzy C means algorithm is more effective than the decision tree as it successfully trains the model to identify the disease by calculating the numerical value based on the severity rating of the symptoms and makes the prediction more accurate by considering the medical history of the patient.

'SYMPTOM'S BASED DISEASES PREDICTION IN MEDICAL SYSTEM BY USING K-MEANS ALGORITHM' co authored by Sathyabama Balasubramanian and Balaji Subramani provides an elucidation to how Hopfield network, LAMSTAR Network and K-Means algorithm can be used to assist the doctors to perform differential diagnosis along with the possible implementation using SOA technique. By using these techniques, it improves the overall speed and increases the accuracy of the algorithm. Especially in large datasets, the LAMSTAR network gave faster and better results. It reduces the effects of misdiagnosis, especially practitioners and students can also easily identify the diseases.

While it is needless to say that e-health can be enormously supplemented by ML, there are quite a few ethical challenges associated with it. The fairness aspect of ML development has to do with supplying representative data for adequate machine analysis. We need to acknowledge that some algorithms may work better or worse for specific groups of populations. Transparency and conflict of interests challenges presuppose appropriate recording and reporting of machine learning model performance metrics. Potential bottlenecks here include the "black-box" nature of ML algorithms, which makes them hard to interpret, and total dependence on the size and features of a dataset and type of available variables.

'Ehealth: Market Potential and Business Strategies' indicates the massive potential of e-health and gives us an insight of its capability to economize the healthcare industry.

However just like any other technological innovation, ehealth has its downsides (Mayo Clinic, 2020). While telehealth has potential for better coordinated care, it also runs the risk of fragmenting health care. Fragmented care may lead to gaps in care, overuse of medical care, inappropriate use of medications, or unnecessary or overlapping care. And further the security of electronic health records is a concern coupled with the sparsity of insurance coverage.

4. System design and Methodology

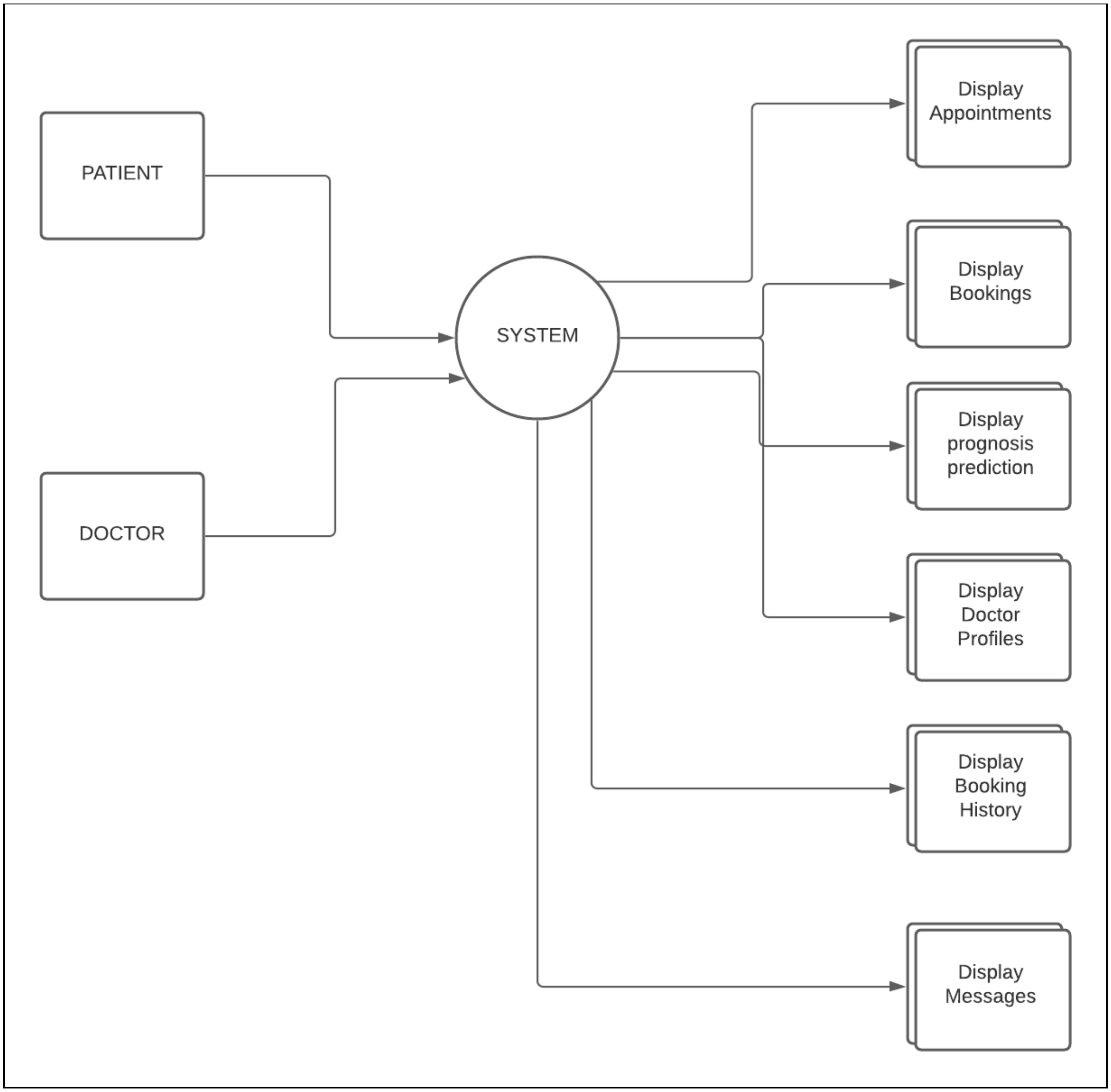
4.1 Business Requirements

The System successfully caters to both patients and doctors wherein;

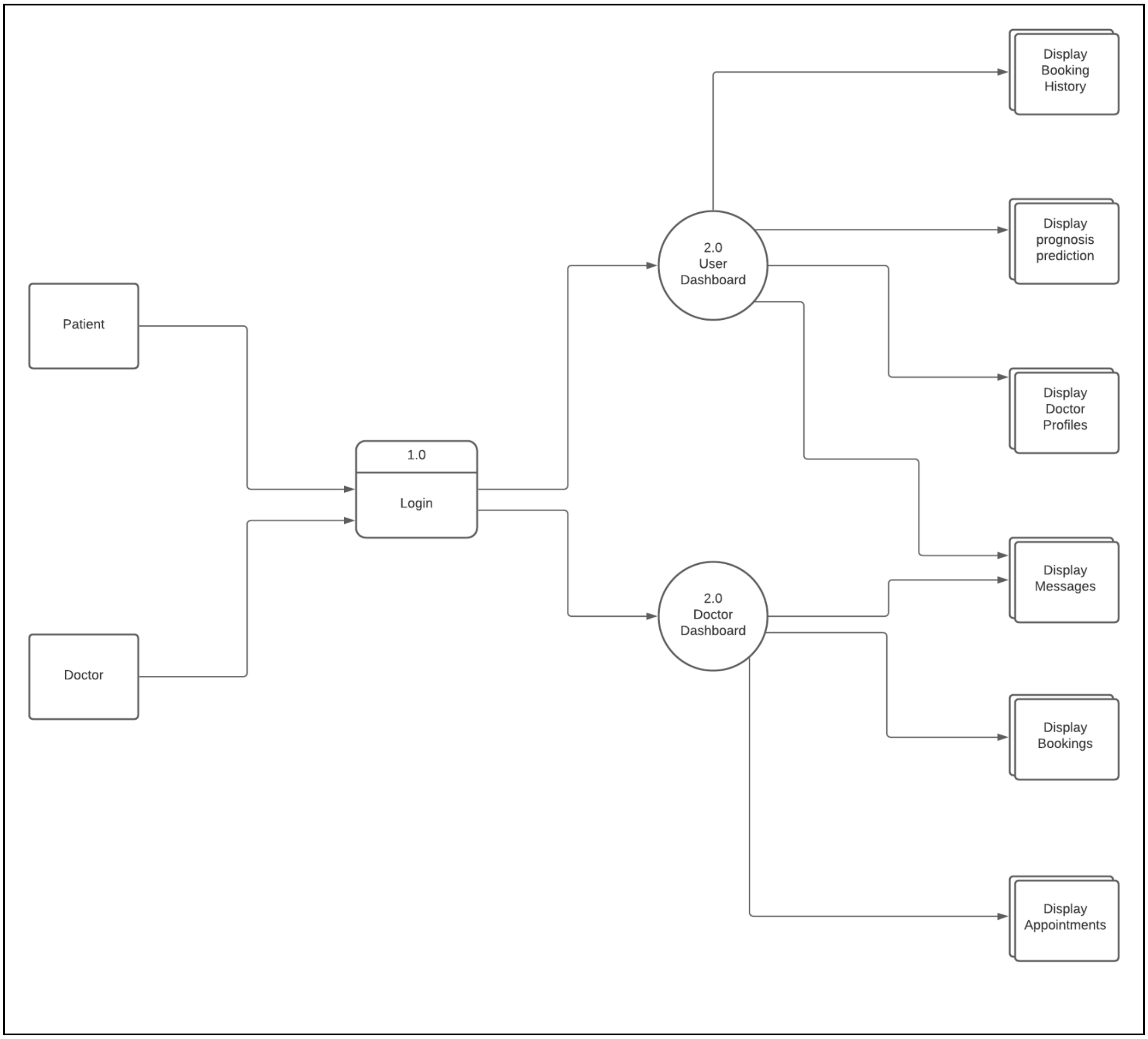
- **Doctors** are able to effectively manage their bookings and schedule their appointments.
- **Doctors** can keep track of the patient's health with the chatbox.
- **Patients** are able to accurately self-diagnose themselves on the basis of their symptoms.
- **Patients** can book appointments with Doctors in their vicinity.
- **Patients** can track their bookings
- **Patients** can communicate with the Doctor with the help of the in app-chat box.

4.2 Data Flow Diagram

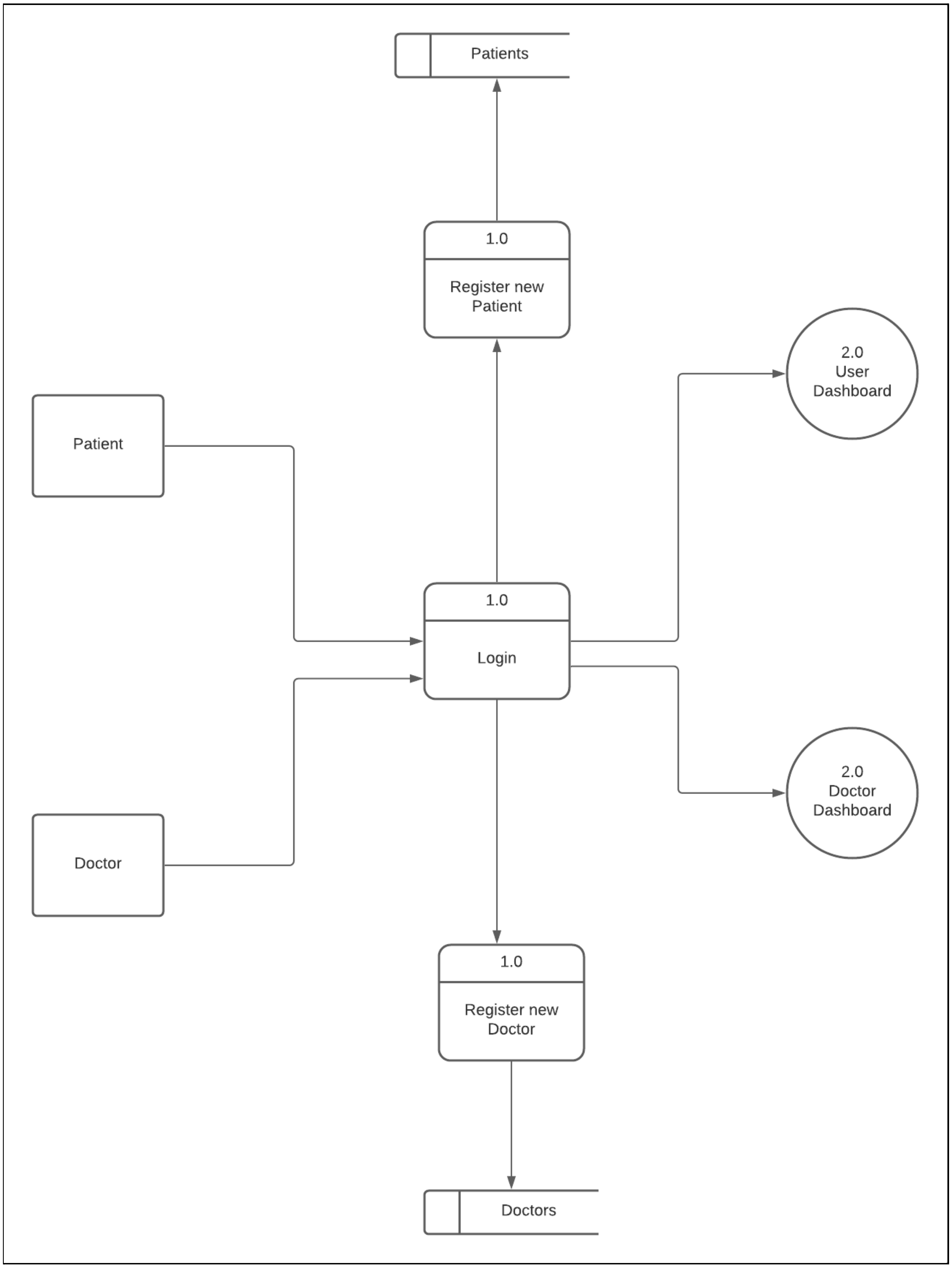
4.2.1 Context Level DFD



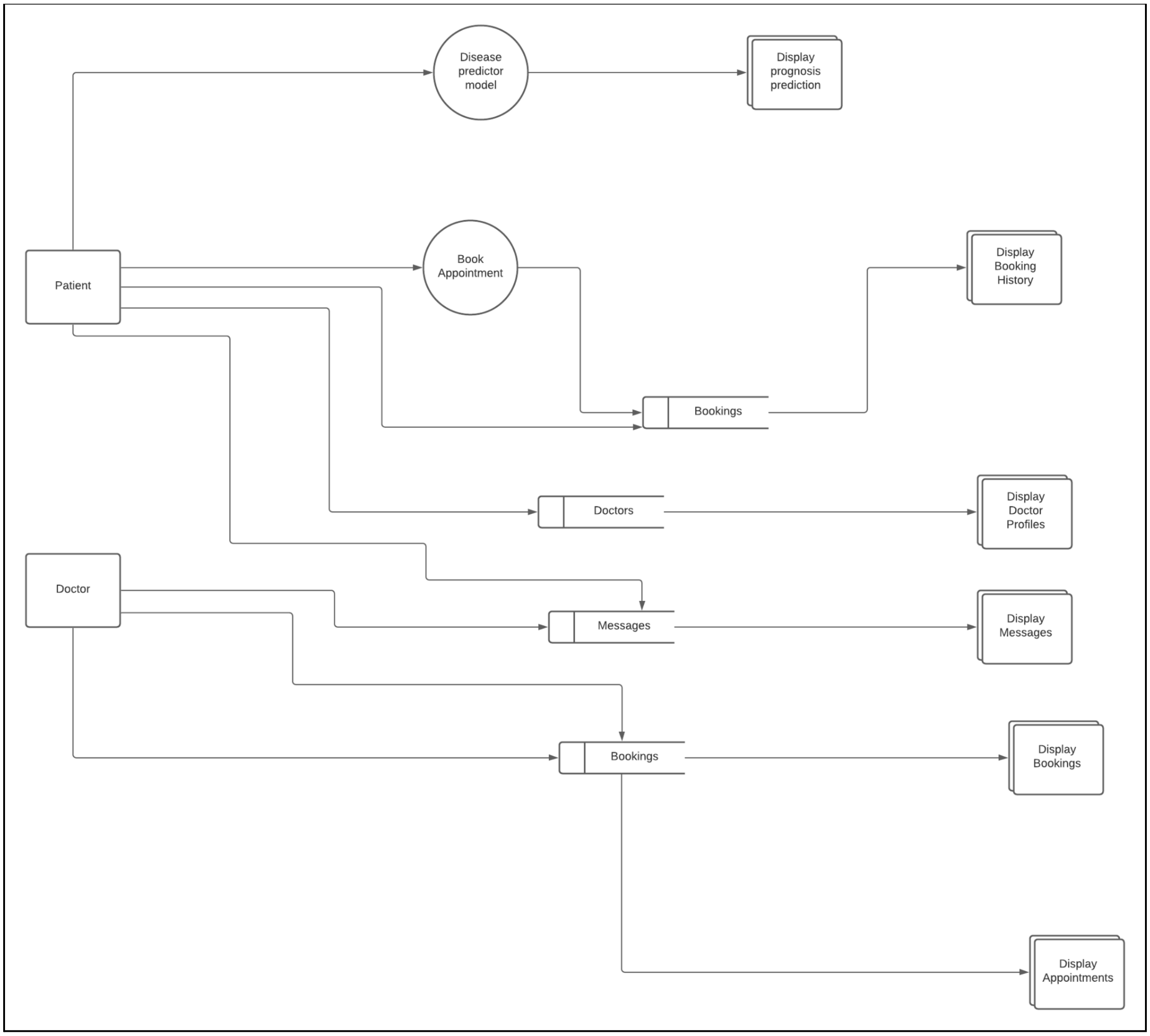
4.2.2 Level-1 DFD Diagram



4.2.3 Level-2 DFD

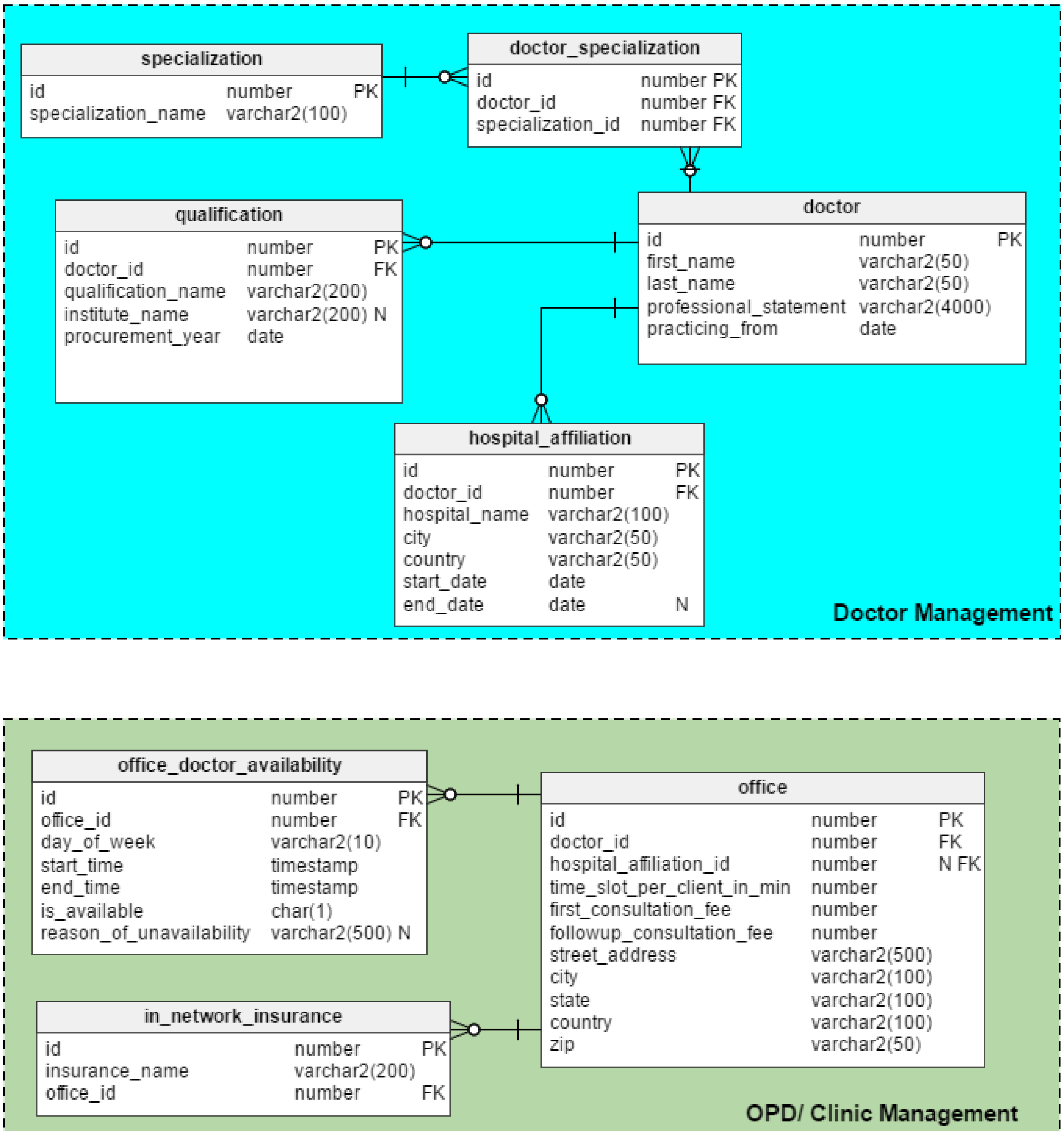


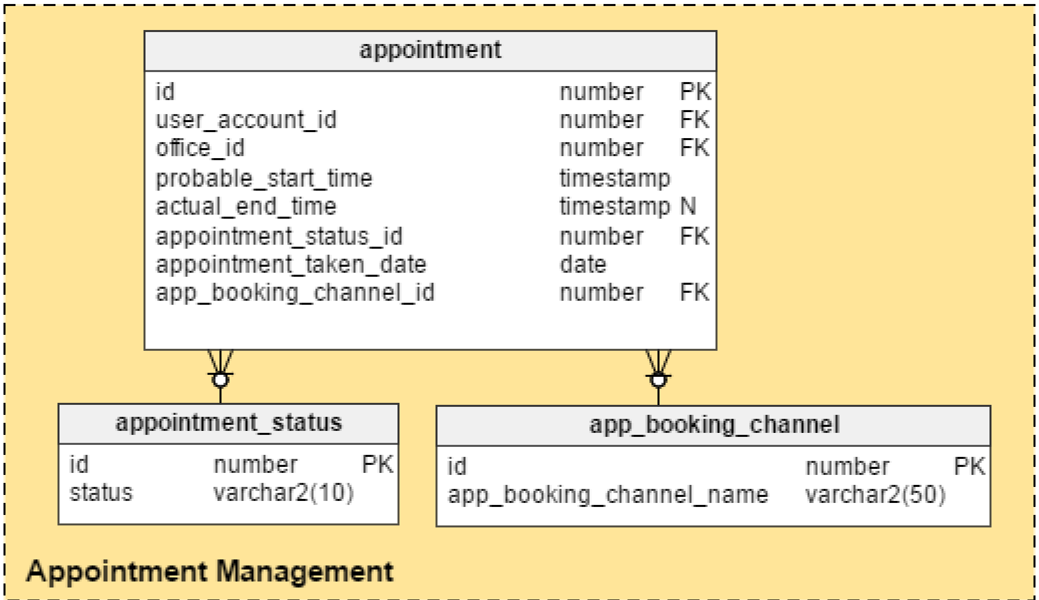
4.2.4 Level 3 DFD



4.3 Database Design

4.3.1 Entity Relationship Diagram




appointment
id number PK
user_account_id number FK
office_id number FK
probable_start_time timestamp
actual_end_time timestamp N
appointment_status_id number FK
appointment_taken_date date
app_booking_channel_id number FK
appointment_status
id number PK
status varchar2(10)
app_booking_channel
id number PK
app_booking_channel_name varchar2(50)
Appointment Management

4.4 System Description

4.4.1 System Structure

Patient:

1. **Register**
    - Users can register with personal details.

2. **Login**
    - User/Patient can login in his personal account using id and password.

3. **Profile**
    - View & update profile

4. **Search Disease**

    - search diseases by name, type or symptoms
    - view disease symptoms
    - view disease description/details

5. **Doctor Details**

    - View profile of the doctors
    - Address of the doctor to visit
    - Book Appointment

6. **Chat**
    - Chat with doctors
    - List All Chats

7. **Book Appointment**
    - Select doctor for the appointment
    - Check if doctor available or any other appointment book at entered time
    - Date of appointment
    - Time of the appointment
    - Charges for the appointment
    - Enter card details for the appointment
    - Wait for the confirmation from the doctor
    - If rejected the amount will be refunded

8. **View Appointment**
    - Check for current, pending, previous or other appointments
    - Book an appointment

Doctor:

Login

- Doctor can login in his personal account using id and password.

My Profile

- view & update profile

View Patient

- View profile of the patients
- Select user/patient from his/her list of patients diagnosed and can chat with selected user/patient.

Prescription

- Add medicine
- Add quantity of intake
- Add time period
- Edit medicines and details
- Delete medicinal record

Chatbox

- Chat with patients
- List All Chats

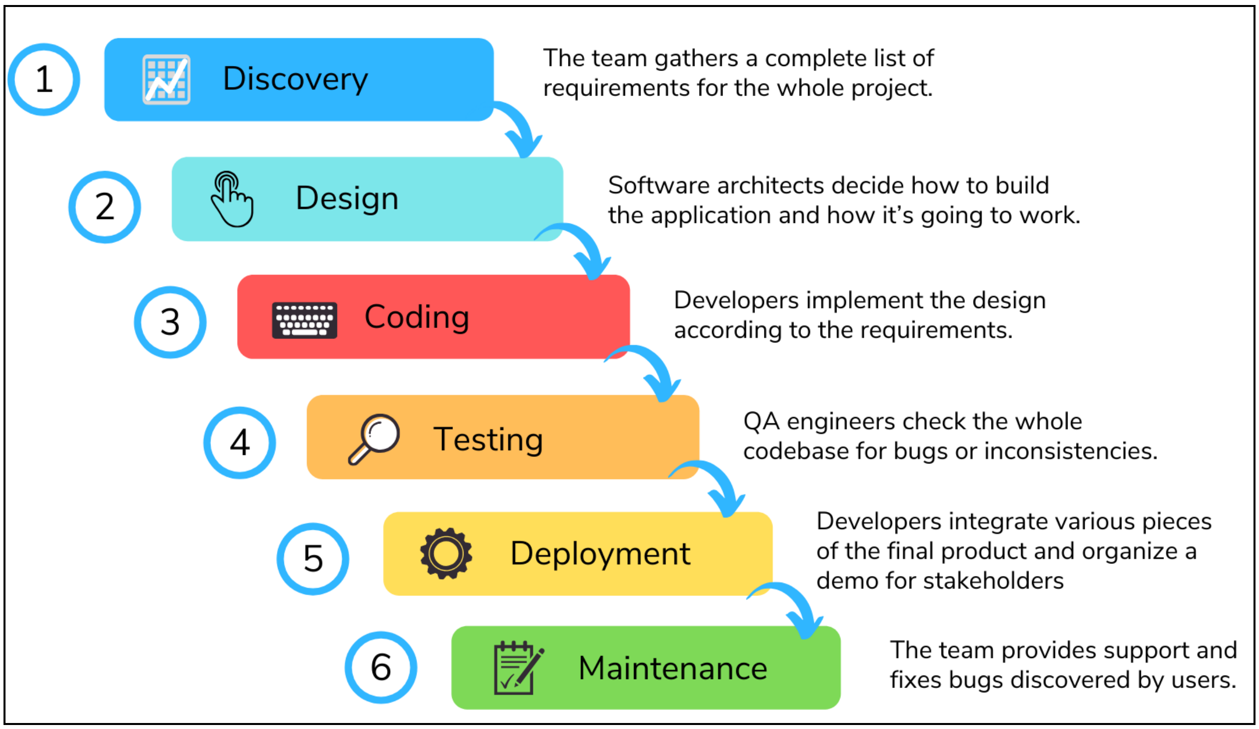
Appointments

- Check for current, pending, previous or other appointments
- Accept user/patient appointments if available

5. System Analysis and Design

5.1 System Development Life Cycle

The system has been developed in accordance with the standards of the waterfall model.



5.2 Tech Stack

5.2.1 Languages

- Java
- Python
- XML
- SQL

5.2.2 Frameworks

- Tensorflow
- TensorflowLite
- Keras
- Scipy
- Firebase API
- Pickle
- Java SDK
- Material Design

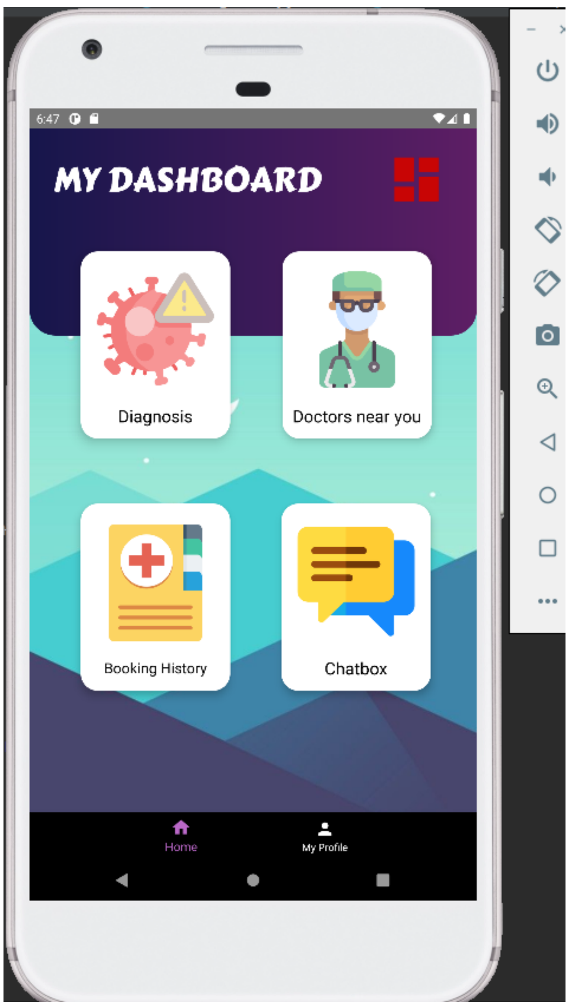
6. Implementation

6.1 Dashboard

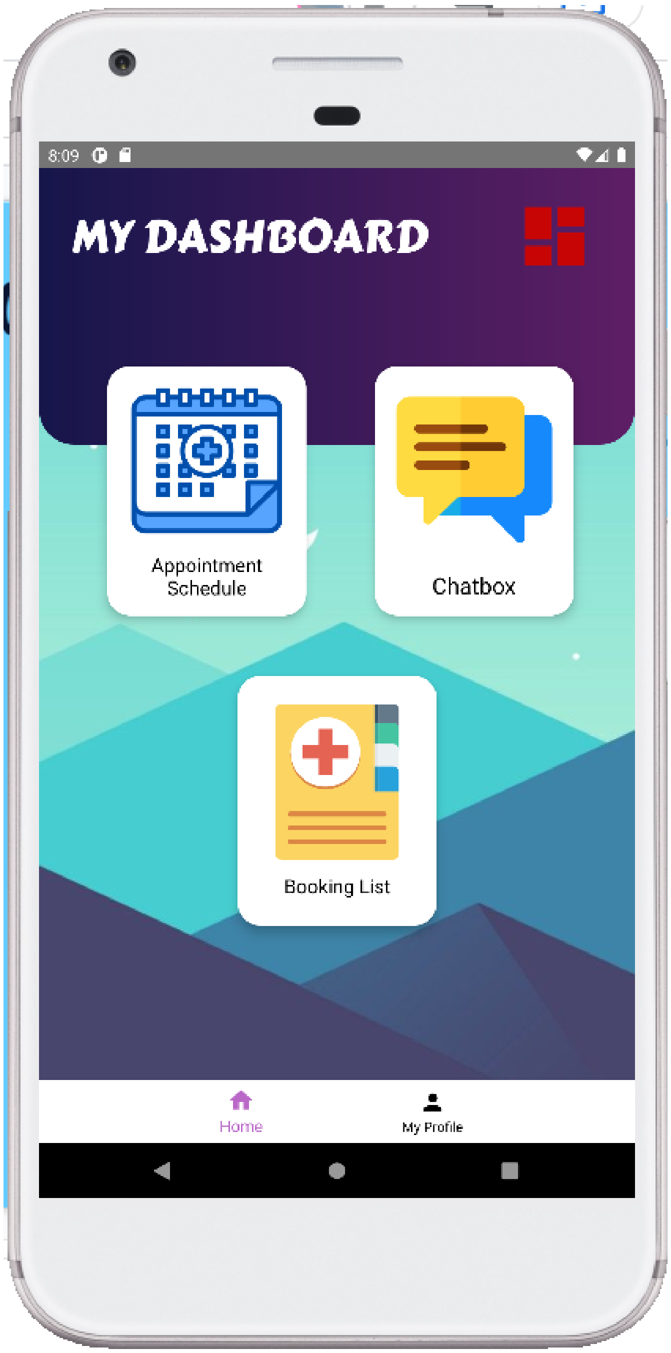
Employs Material Design with the help of card view widgets accompanied by customized layout files.

Relies on the user code retrieved from the intent extras bundle to display the appropriate dashboard(1- Doctor, 2- Patient).

6.1.1 Patient Dashboard



6.2.2 Doctor Dashboard



6.2 Disease Prediction System

System makes use of deep learning to train a model that accurately predicts diseases based on their symptoms.

Following are the steps taken to train and validate our model successfully:-

- Data preprocessing
- Feature Engineering
- Train model on training data
- Test Model on testing data
- Deploy model into android studio by converting model to tensorflowLite

Model Summary

```
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 74)                9842
_________________________________________________________________
dense_1 (Dense)              (None, 60)                4500
_________________________________________________________________
dense_2 (Dense)              (None, 41)                2501
=================================================================
Total params: 16,843
Trainable params: 16,843
Non-trainable params: 0
_________________________________________________________________
```

Training Model

```
from sklearn.utils import shuffle
x_train,y_train= shuffle(X_train,Y_train)
y_train=y_train.reshape(-1,1)
y_train.shape
```

```
[7]: (4920, 1)
```

```
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Activation
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.metrics import sparse_categorical_crossentropy
```

```
model= Sequential([
    Dense(units= 74, input_shape=(132,),activation= 'relu'),
    Dense(units= 60,activation='relu'),
    Dense(units=41, activation='softmax')

])
```

```
model.compile(optimizer= Adam(learning_rate=0.0001),loss='sparse_categorical_crossentropy',metrics=['accuracy'])
```

```
model.fit(x=x_train,y=y_train,validation_split=0.3,batch_size=30,epochs=30,verbose=2)
```

```
Epoch 1/30
115/115 - 0s - loss: 3.6517 - accuracy: 0.0526 - val_loss: 3.5734 - val_accuracy: 0.1497
Epoch 2/30
115/115 - 0s - loss: 3.4812 - accuracy: 0.2300 - val_loss: 3.3806 - val_accuracy: 0.3679
Epoch 3/30
115/115 - 0s - loss: 3.2500 - accuracy: 0.5200 - val_loss: 3.1133 - val_accuracy: 0.6321
Epoch 4/30
115/115 - 0s - loss: 2.9405 - accuracy: 0.6800 - val_loss: 2.7716 - val_accuracy: 0.7297
Epoch 5/30
115/115 - 0s - loss: 2.5608 - accuracy: 0.8026 - val_loss: 2.3698 - val_accuracy: 0.8604
Epoch 6/30
115/115 - 0s - loss: 2.1393 - accuracy: 0.9036 - val_loss: 1.9439 - val_accuracy: 0.9593
Epoch 7/30
115/115 - 0s - loss: 1.7157 - accuracy: 0.9628 - val_loss: 1.5288 - val_accuracy: 0.9776
Epoch 8/30
```

Model Validation

```
score=model.evaluate(x=x_test,y=y_test,verbose=0)
```

```
print('Test Loss: ',score[0])
print('Test Accuracy: ',score[1])
```

```
Test Loss:  0.04063546657562256
Test Accuracy:  1.0
```

Deploy Model

```python
import tensorflow as tf
```

```python
converter = tf.lite.TFLiteConverter.from_keras_model(model)
converter.optimizations = [tf.lite.Optimize.DEFAULT]
tflite_quant_model = converter.convert()
```

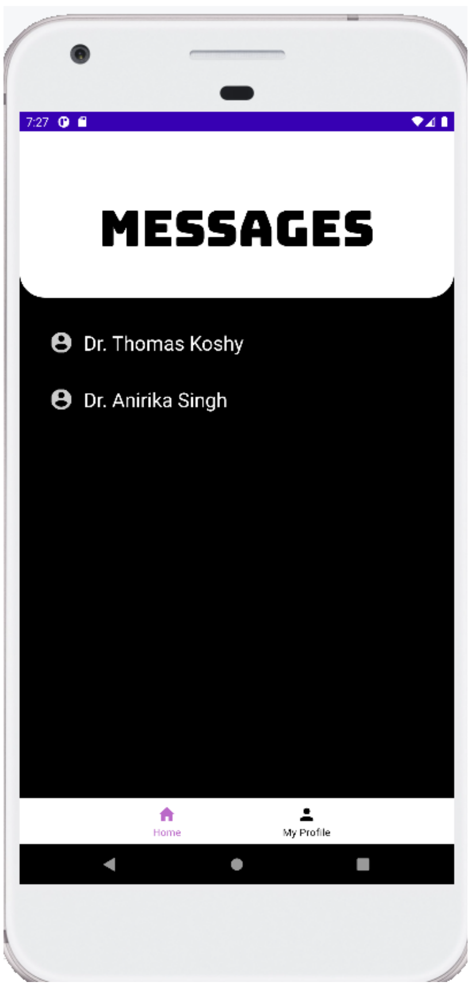
```
WARNING:tensorflow:From C:\Users\Niko\Anaconda3\envs\ML_env\lib\site-packages\tensorflow\python\training\tracking\tracking.p
y:111: Model.state_updates (from tensorflow.python.keras.engine.training) is deprecated and will be removed in a future vers
ion.
Instructions for updating:
This property should not be used in TensorFlow 2.0, as updates are applied automatically.
WARNING:tensorflow:From C:\Users\Niko\Anaconda3\envs\ML_env\lib\site-packages\tensorflow\python\training\tracking\tracking.p
y:111: Layer.updates (from tensorflow.python.keras.engine.base_layer) is deprecated and will be removed in a future version.
Instructions for updating:
This property should not be used in TensorFlow 2.0, as updates are applied automatically.
INFO:tensorflow:Assets written to: C:\Users\Niko\AppData\Local\Temp\tmp95d_wc05\assets
```

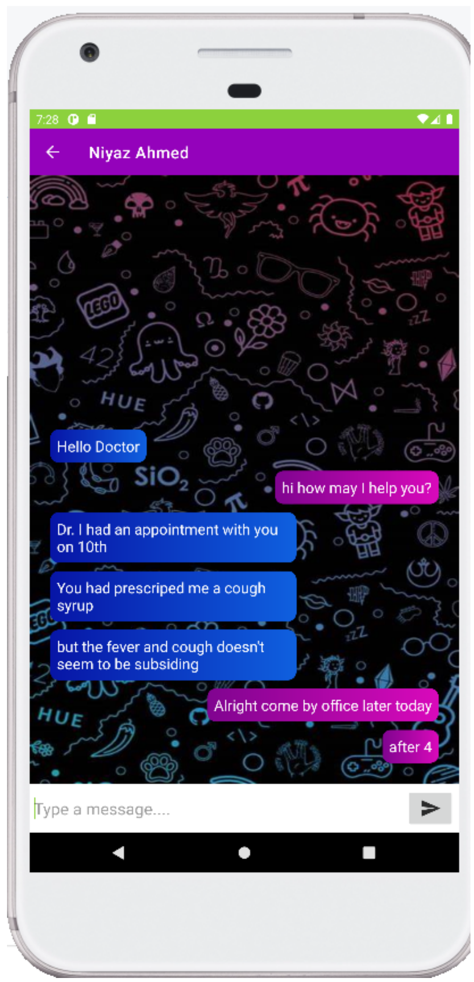
```python
with open('models/dpredictor.tflite','wb') as file:
    file.write(tflite_quant_model)
```

6.3 ChatBox

Messages are stored in the firebase realtime database along with the messenger ID and receiver ID.

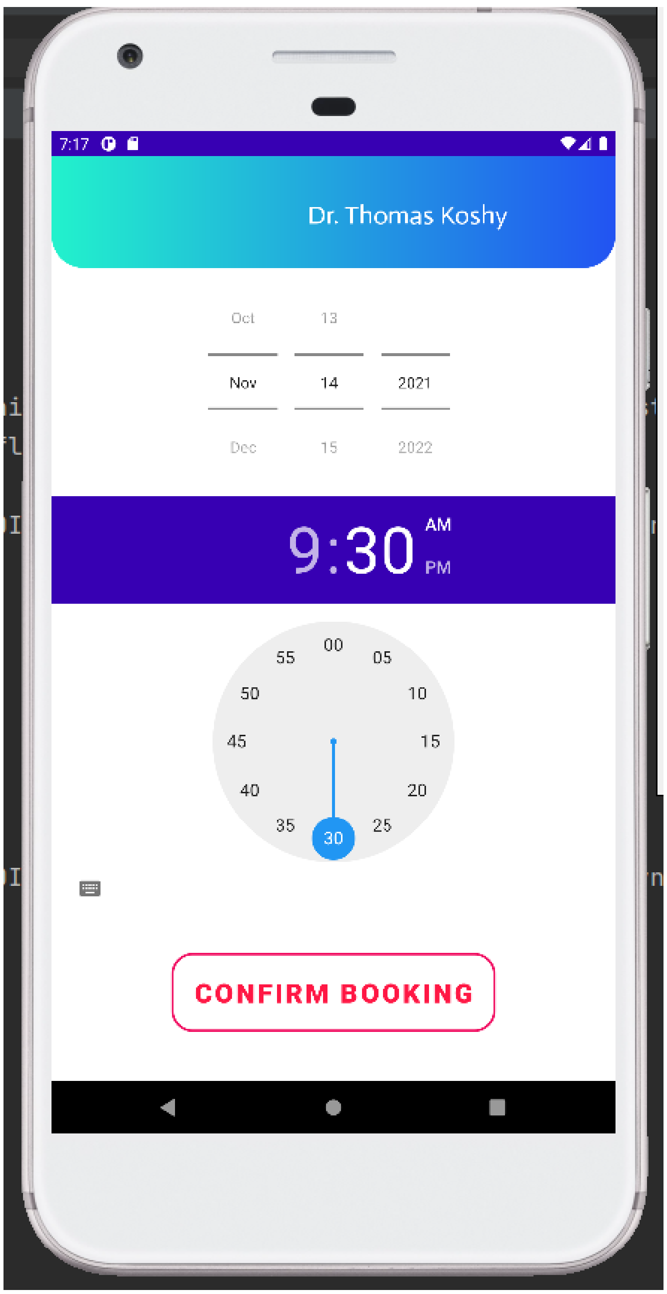
Two XML layout files are created for messenger and receiver messages respectively along with the recycler view adapter class to display the messages retrieved from the database.





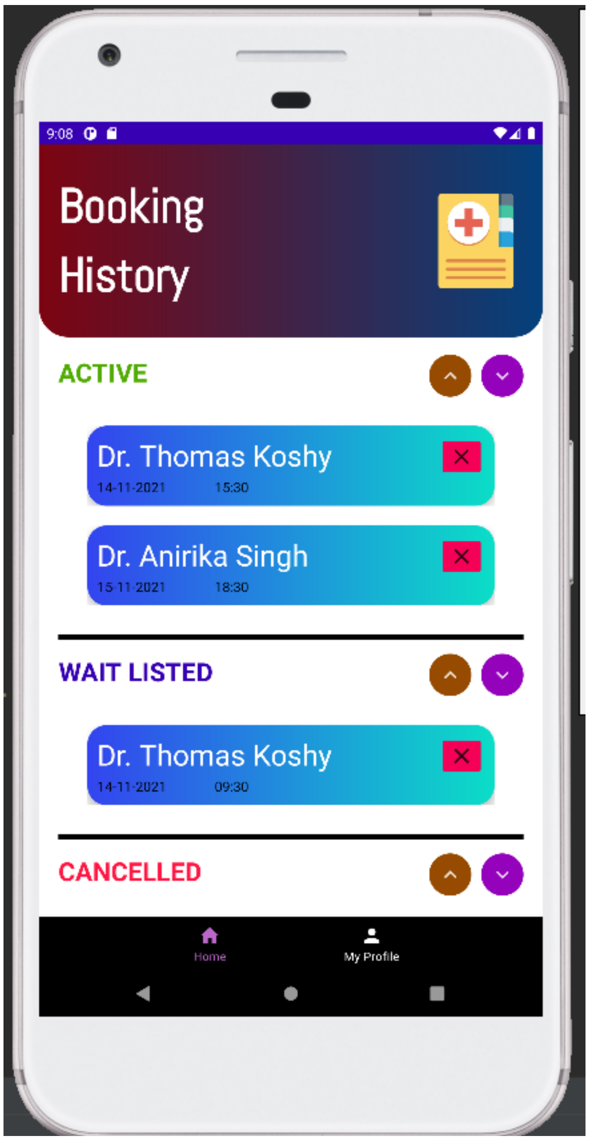
6.4 Booking Appointments

Users can book appointments with the help of the time picker and date picker dialog while adhering to the constraints put up by the doctor.



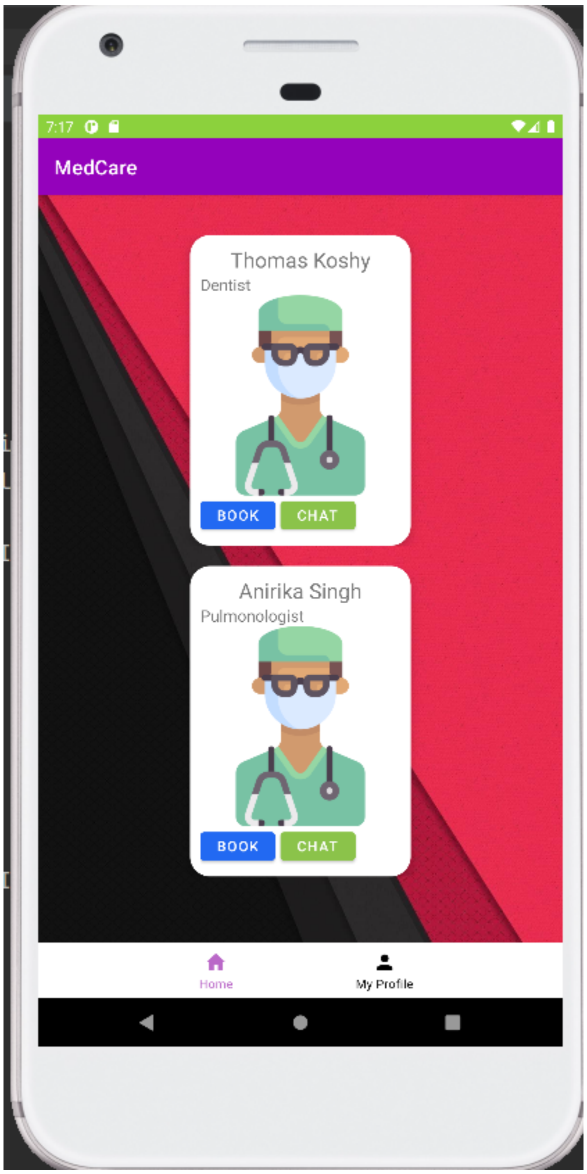
6.5 Booking History

Users are able to actively view the status of bookings, i.e., ***Active, Waitlisted or cancelled.***



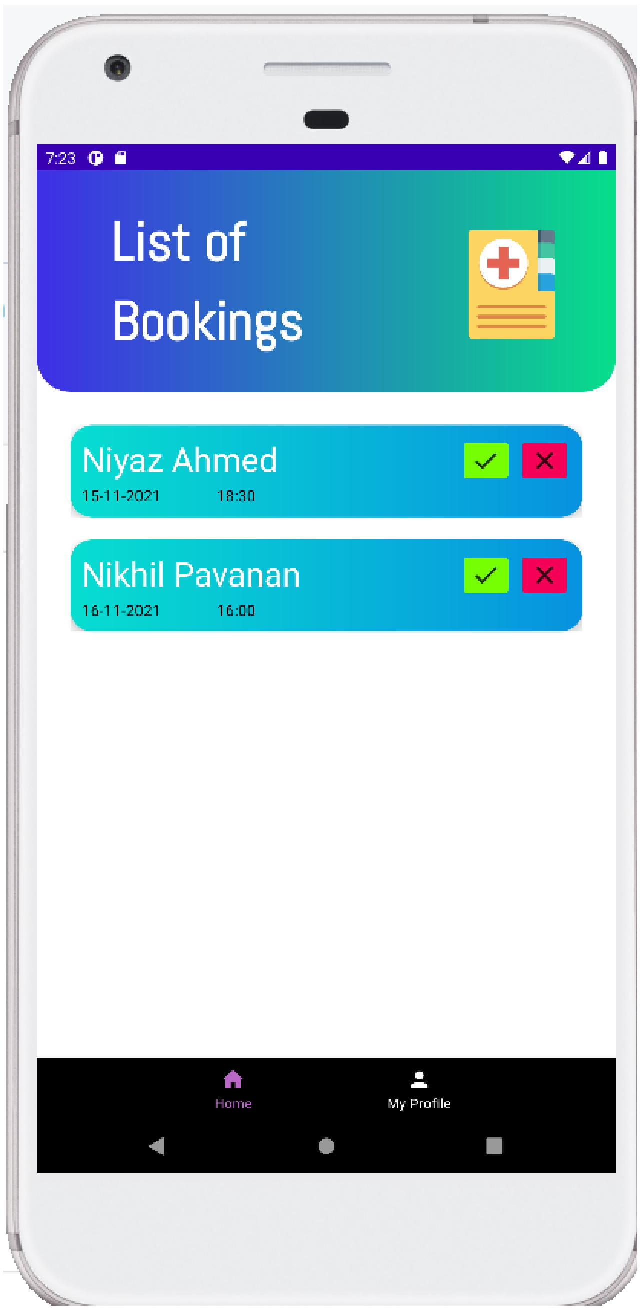
6.6 View Doctor Profiles

Retrieves data from the database to provide information of all the doctors in your region accompanied with additional button widgets that successfully redirects you to either the 'booking' or 'send new message' activity.



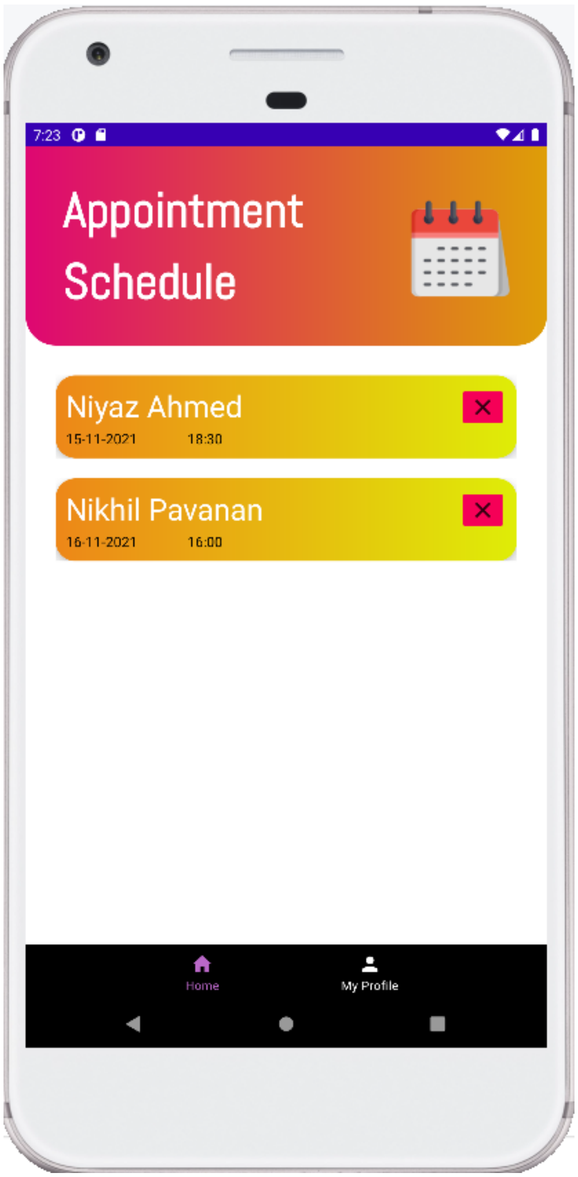
6.7 List of Bookings

Display the list of bookings retrieved from the database with the status label '*waitlisted*' and further wait for the doctor's instructions on whether to confirm bookings or cancel.



6.8 Appointment Schedule

Doctors can track their appointments and cancel them if required.



7. Future Scope

1. Introduce healthcare chatbot that engages and provides solutions to users.
2. Include Immediate Medical Attention/ Emergency feature that employs Distance Matrix API to pinpoint the location of the nearest hospital and notifies the hospital to send medical aid via sms automation.
3. Prescription tracker
4. Online pharmacy that allows users to order medicines and other medical supplies online.
5. Establish a video conferencing system that facilitates virtual appointments.
6. Incorporate Convolutional Neural Networks to analyse and process medical images online.

8. References

Nilsson, N. J. (1998). Introduction machine learning - home - stanford artificial ... INTRODUCTION TO MACHINE LEARNING. Retrieved October 22, 2021, from https://ai.stanford.edu/~nilsson/MLBOOK.pdf.

Arumugam, K., Naved, M., & Shinde, P. (2021). Multiple disease prediction using Machine learning algorithms. Redirecting. Retrieved October 22, 2021, from https://doi.org/10.1016/j.matpr.2021.07.361.

Sathyabama, S., & Subramani, B. (2014). Symptom’s based disease prediction in the medical system by using K-Means algorithm. Retrieved October 22, 2021, from http://warse.org/pdfs/2014/ijacst13322014.pdf.

Hubel, David H., and Torsten N. Wiesel. “Receptive fields and functional architecture of monkey striate cortex.” The Journal of physiology 195.1 (1968): 215–243.

Sarvamangala, D.R., Kulkarni, R.V. Convolutional neural networks in medical image understanding: a survey. Evol. Intel. (2021). https://doi.org/10.1007/s12065-020-00540-3